Motivation

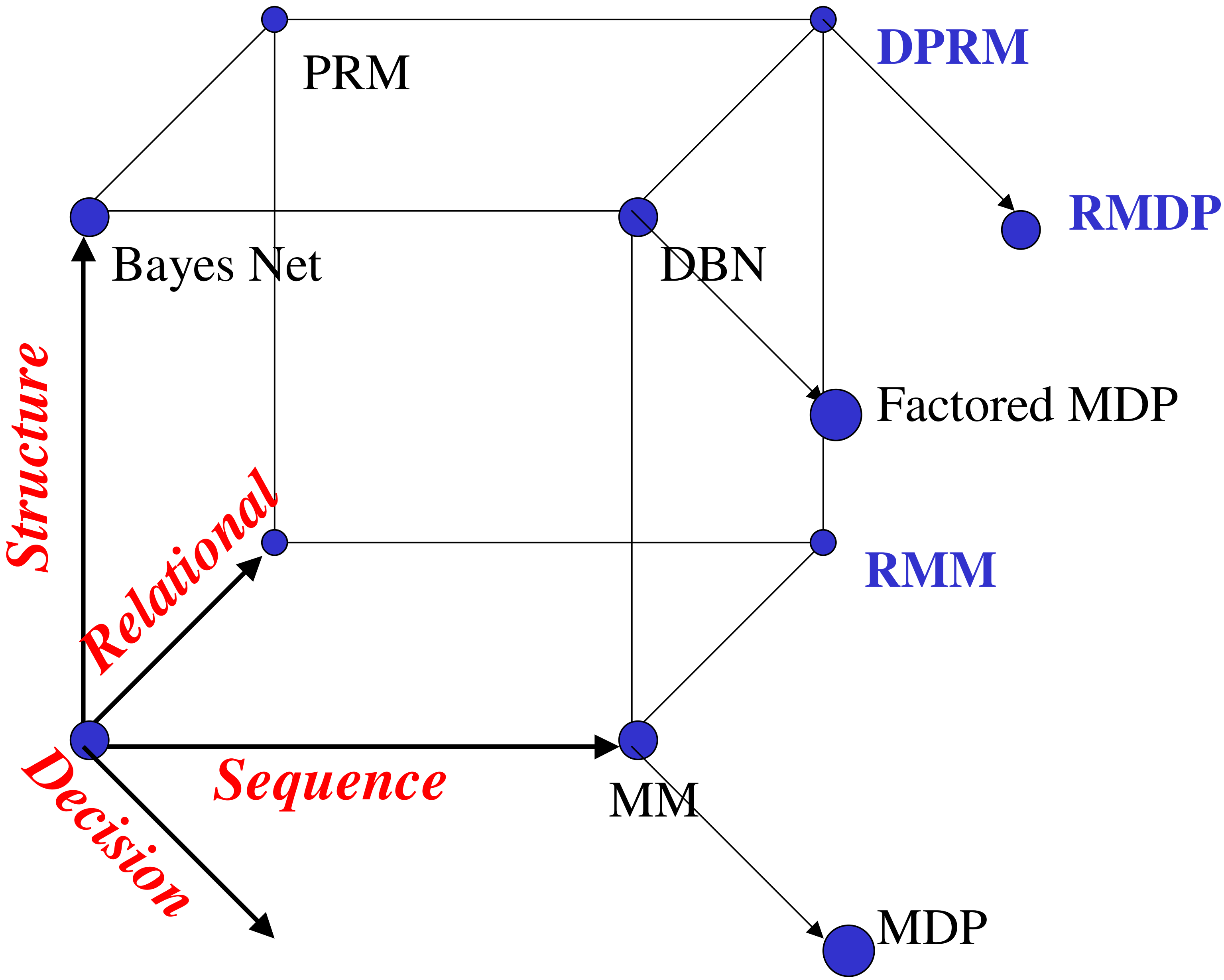
- Models for Uncertain Sequential Data
  - Markov Models
  - Dynamic Bayes Nets (DBNs)
  - (Factored) Markov Decision Processes (MDPs)
- Powerful + Ubiquitous, but Lacking
  - Static set of state variables & relationships
  - Propositional – no notion of object & relations
  - No quantification
- Spurred by Recent Advances (e.g. OOBNs, PRMs)…
  - Combining ideas from FOL and Probabilistic Graphical Models

$\Rightarrow$ Relational Markov Models

$\Rightarrow$ Dynamic PRMs

$\Rightarrow$ Relational MDPs

Overview

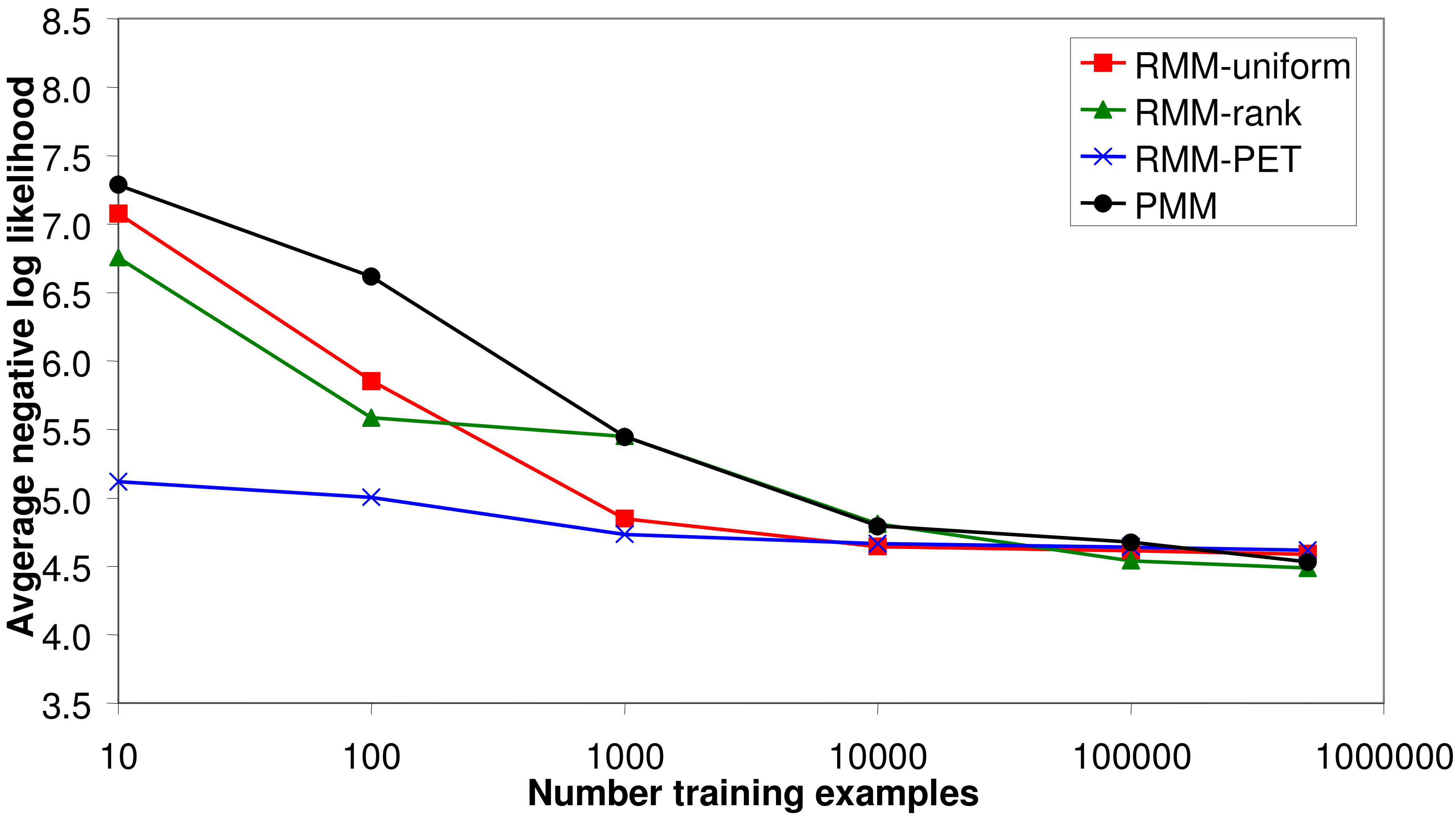


PRM
DPRM
RMDP
Bayes Net
DBN
Factored MDP
Structure
Relational
RMM
Decision
Sequence
MM
MDP

Relational Markov Models

- In ordinary MM, each state is trained independently
  - Abundant training data for one state cannot improve prediction at another state
  - Large state models require vast training data
- Relational MMs exploit relational structure in domain
  - Given abstraction hierarchy over each data type…
  - Structure enables state generalization…
  - Combats data sparseness with shrinkage
    - ⇑ **Weighting when abstractions are more specific**
    - ⇑ **Weighting when training data is abundant**
- Learned RMMs outperform PMMs

Learning RMMs *vs.* Propositional MMs



(Gazelle ecommerce dataset)

Dynamic PRMs

- Aka Relational DBNs
- Dynamic object creation
- Learning
  - Modified version of PRM learner
- Inference
  - Modified version of particle filters

RMDP Objectives

- Define different classes of objects
- Possible relations between objects
- Action schemata

- Semantics in terms of a ground MDP

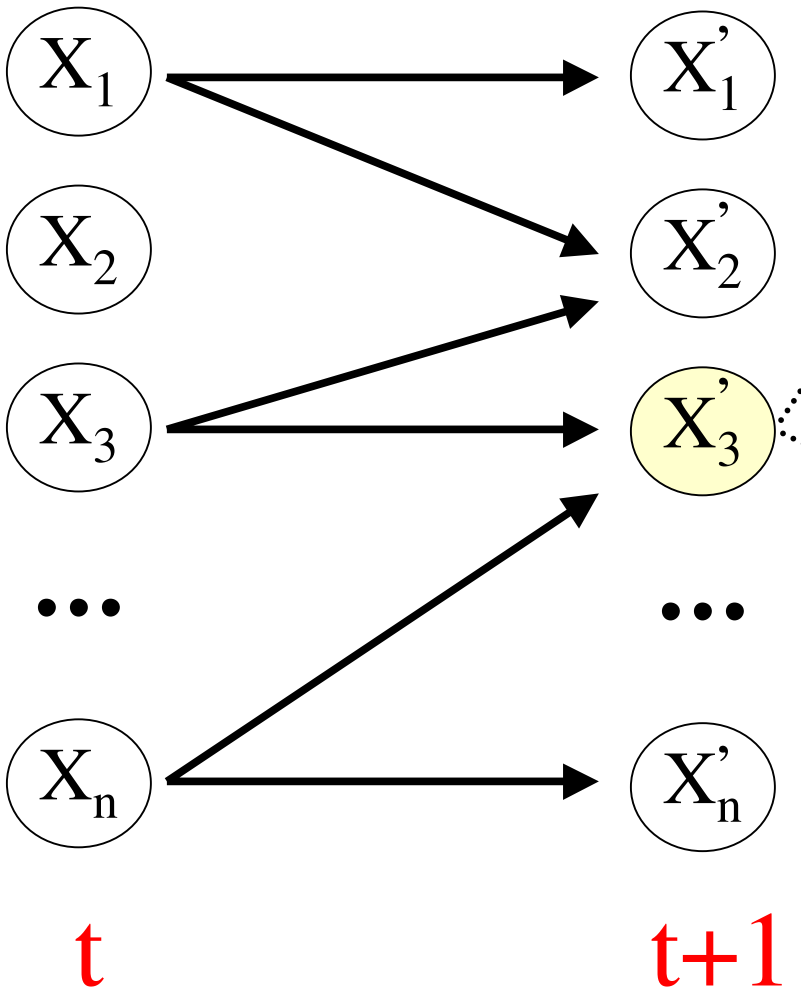
- Benefits
  - Convenient specification of complex domains
  - Exploit structure for faster policy construction
  - Handle domains w/ dynamic relations, object creation / destruction

Review: Factored MDP

- Space of states: S
  - Characterized by variables $X = \{X_1, \ldots, X_n\}$
- Set of actions: A
  - Each specified using a DBN
- Transition function: $P(s' \mid s, a) \rightarrow [0, 1]$
- Reward function: $R(s, a) \rightarrow \mathcal{R}$

- Objective: compute a policy $\pi: S \rightarrow A$
  - Maximizing discounted reward

Review: DBN Spec. of an Action



X_3	X_n	X'_3
F	F	.1
F	T	.6
T	F	.9
T	T	1.0

CPT
(often specified as an ADD)

Relational MDPs

- Specify classes of objects and possible relations
- States characterized by relational interpretations
  - Instead of a set of propositions: $X = \{X_1, \ldots, X_n\}$

- Actions are schematized
  - May change the set of objects, relations between them
    - E.g. a manufacturing mill which produces new objects
    - Or a robot's motion which discovers new objects

Improved Value & Policy Iteration

- Use relational structure to aggregate states
  - Factor state space via homeomorphisms
  - Conditional irrelevance of wffs
  - Augmented operator-graph analysis

- Update ***multiple*** states with each Bellman backup

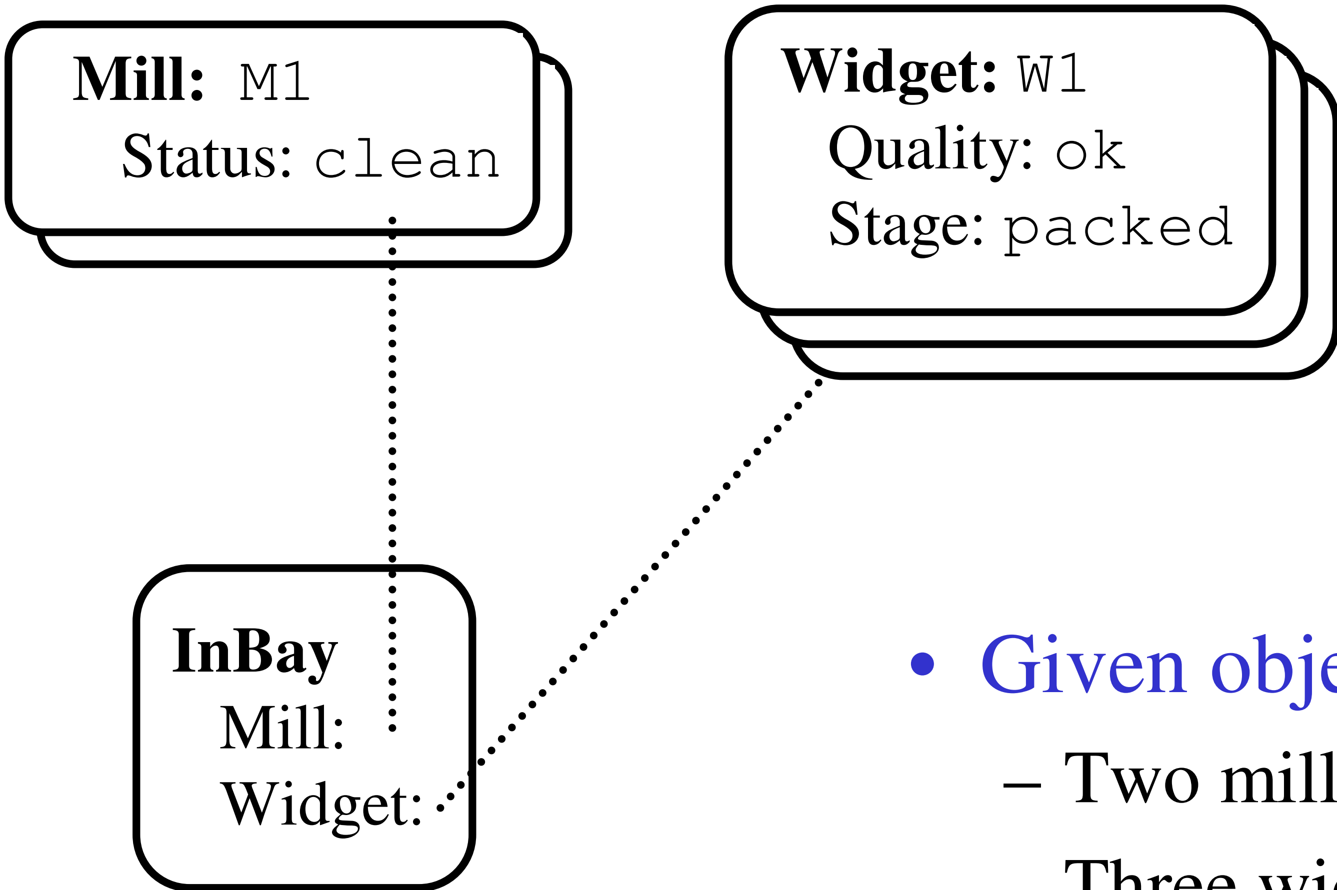
$$V(s) = \max_{a} \left[R(s, a) + \gamma \sum_{s'} P(s' \mid s, a) V(s') \right]$$

Dynamic Object Creation…

Simple RMDP Example

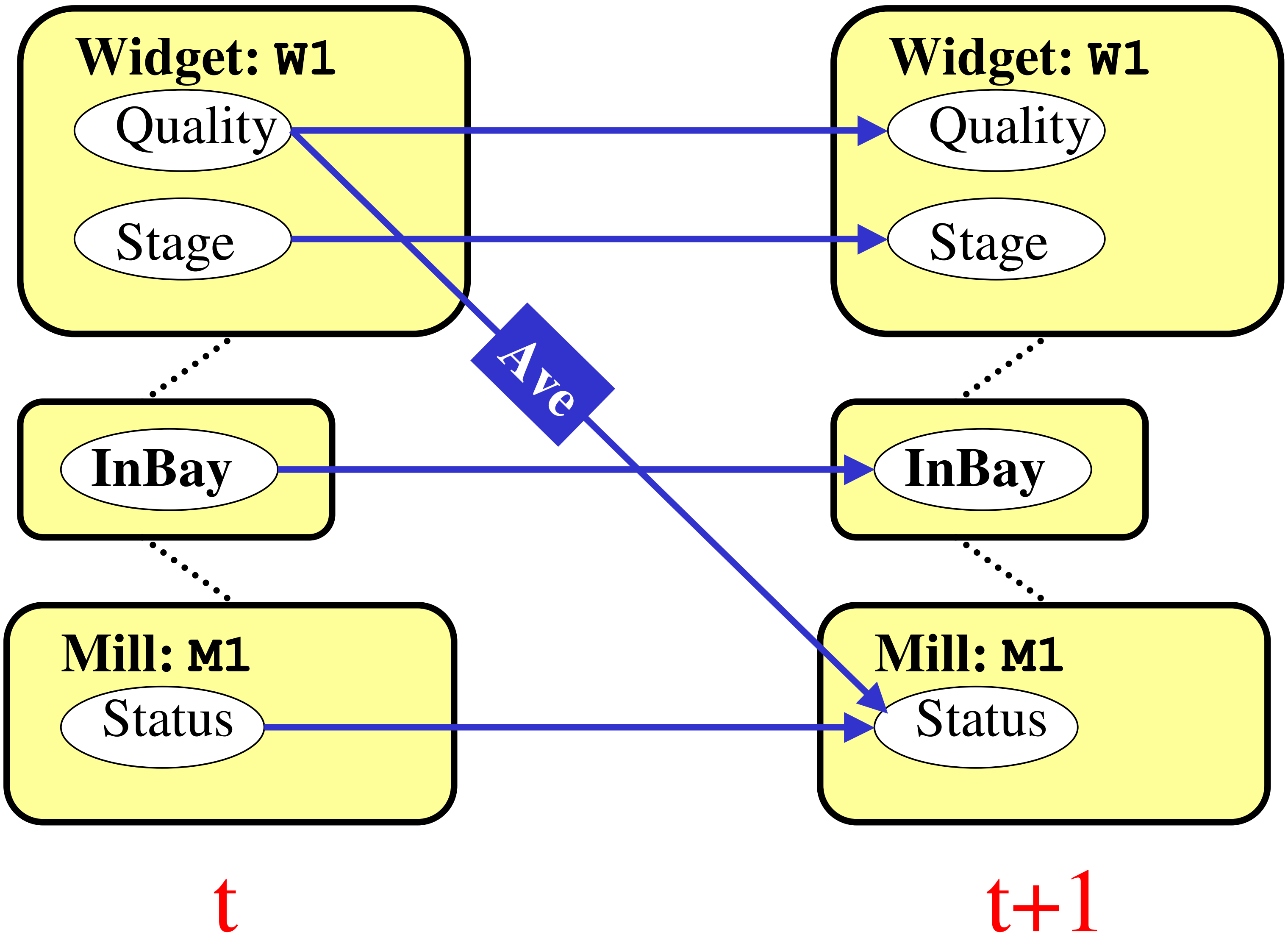
- Two Types of Objects w/ Boolean Attributes
  - Mill (status) *e.g., dirty / clean*
  - Widget (quality, stage) *e.g., defective / ok; ready / packed*
- One Type of Relation Possible
  - InBay(mill, widget)
- Three Actions
  - Process(M) *Create a new widget (maybe defective)*
  - Pack(W) *Pack (good) widget, clearing mill bay*
  - Recycle(W) *Clear mill bay*
- Large State Space
  - Suppose m mills, w widgets => $2^{(m+2w+mw)}$ states
  - E.g., 4 mills, 10 widgets => 10^{19} states

Example: Relational Skeleton

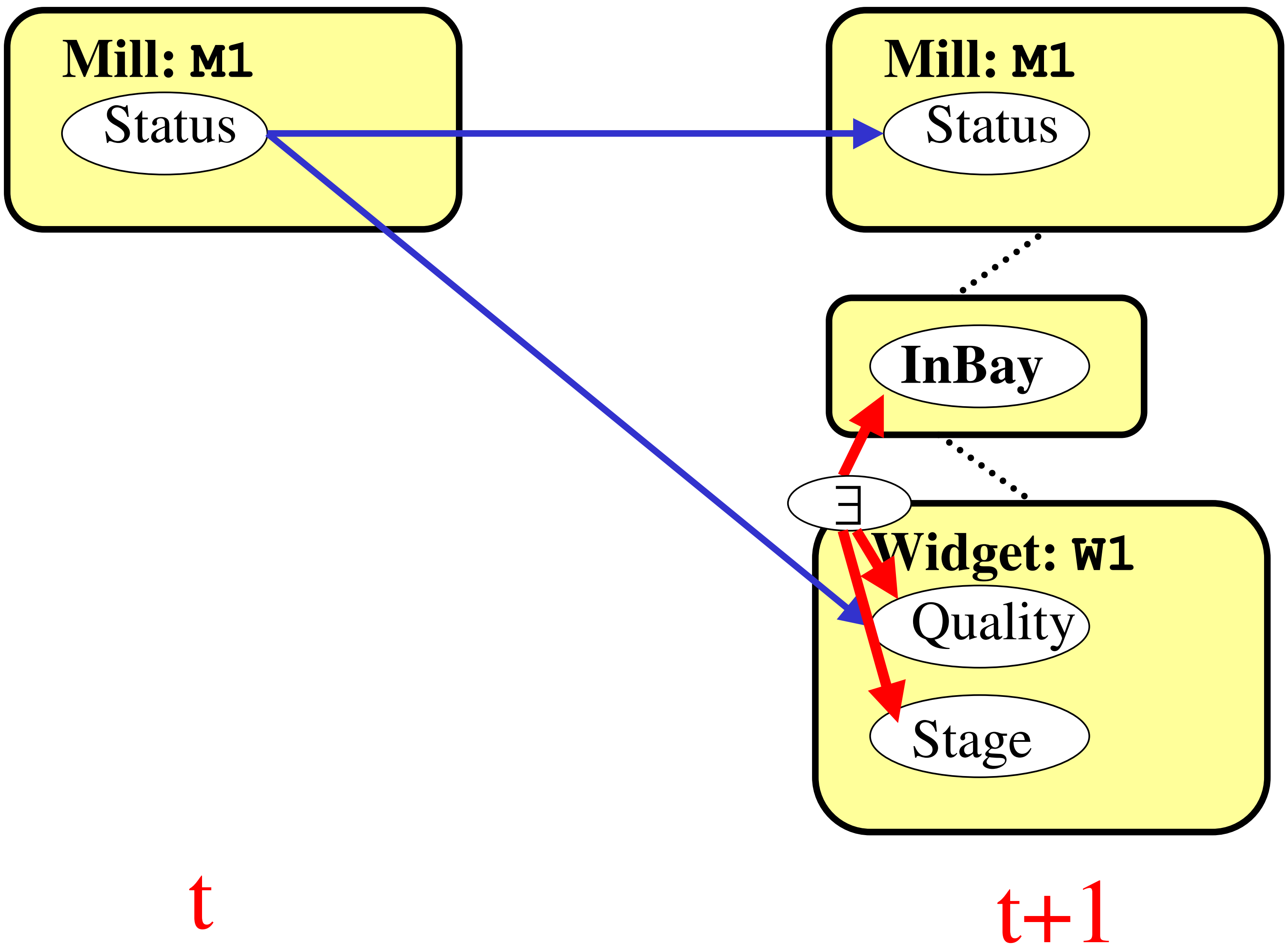


- Given objects
  - Two mills
  - Three widgets

Example Action: Pack(W)



Example Action: Process(M)



Related Work

- PRMs [Friedman *et al*.]
- Factorial HMMs [Ghahramani & Jordan]
- OOBNs [Pfeffer *et al*.]
- Use of hierarchy in reinforcement learning
- *Etc*.

Conclusion

- RMDPs allow easier modeling of complex domains
- Exploit structure for faster policy construction
- Dynamic relations, object creation / destruction

Planning and Execution Under Uncertainty

Daniel S. Weld, University of Washington

Objectives & Innovations

- Combine ideas from MDPs & relational logic
  - Convenience & expressiveness
  - Exploit structure to speed policy construction
  - Dynamic objects / relations
- Formalize unified agent architecture
  - Define interleaved planning & execution …

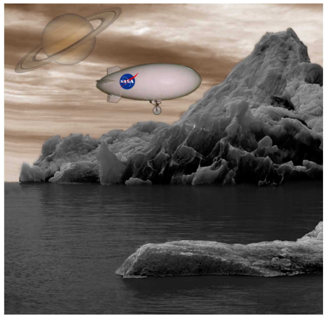
    … as lazy evaluation of contingent planning

Relational Markov Decision Processes

- Specify classes of objects and possible relations
- States characterized by relational interpretations
  - Instead of a set of propositions
- Actions are schematized
  - May change the set of objects
  - May change the relations between objects

NASA Relevance

- Uncertainty is ubiquitous in rover context ⟶ MDPs
- Efficient processing crucial given processor constraints
- Ability to handle novel objects, changing relations



Accomplishments

- Grant Initiation *[Feb 02]*
- Paper on Relational Markov Models *[KDD 02]*
- Definition: Relational MDP *[July 02]*

Milestones *[one year grant]*

- Paper on RMDPs *[Oct 02]*
- Experiments on new objects *[Jan 03][*

Example Action: Process(M)